

DETAILED ACTION

1. This action is responsive to the following communication: Request for Continued Examination filed 06/14/2010.

Continued Examination Under 37 CFR 1.114

2. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 06/14/2010 has been entered.

EXAMINER'S AMENDMENT

3. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it **MUST** be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Malgorzata Kulczycka (Reg. No. 50,496) on 09/02/2010.

This application has been amended as follows:

Claim 2. (Currently Amended) A method comprising the computer-implemented steps of:
automatically generating a consistent user interface for an application program by assisting a user with building an (HTML) hypertext markup language (HTML) user interface page by:
receiving one or more business objects that each define a user action for the application program;
receiving one or more metadata elements for dynamic content generation and defining parameters for the user actions of the one or more business object;

invoking a controller that is communicatively coupled to one or more actions, one or more widgets, and one or more panels;

receiving a first user request from the user through a browser used to interact with the application program and dispatching the user request to one or more of the actions;

based on a second user request received from the user through the browser, selecting a first panel from the one or more panels and including the first panel in the HTML user interface page;

wherein the first panel is dynamically generated using the one or more widgets and comprises a programmed template with a user-interface content;

wherein the user-interface content comprises one or more [any] of user controls, programmable buttons and action buttons;

the controller determining which of the one or more actions is responsible for acting on the first user request;

obtaining, using the one or more actions, one or more parameter values from the business objects and dynamically manipulating the one or more parameter values;

using the one or more actions and the business object parameter values, selecting a first widget from the one or more widgets;

associating the first widget with the first panel selected from the one or more panels, wherein the selected first widget is arranged into a specified dynamic layout within the first panel; and

generating the specified dynamic layout, at runtime, and presenting to the user, a user-interface utility comprising the HTML user interface page that includes the first widget arranged into the specified dynamic layout within the first panel; wherein the first widget has the capability of representing properties of the business objects as HTML elements; wherein the method performed by one or more processors.

Claim 5. (currently amended) A method as recited in Claim 2, wherein receiving one or more business objects that define functions of the application program comprises receiving an [XML] extensible markup language (XML) file that defines the business objects and one or more of the parameters for the business objects.

Claim 8. (currently amended)
A method as recited in Claim 2, further comprising the steps of:

Art Unit: 2179

receiving user input in a field of the user interface that is associated with the widget, wherein the user input is received in HTML elements of an [HTTP] hypertext transfer protocol (HTTP) request from a browser;

converting the user input from the HTML elements into one or more programmatic objects having an appropriate data type for use by the application program.

Claim 11. (currently amended)

A method as recited in Claim 2, wherein the step of generating an HTML user interface page that includes the panel further comprises generating an HTML user interface page that includes one or more of [JSP] java server pages (JSP) files, static HTML elements, style sheets, or images.

Claim 13. (Currently Amended) A computer-readable volatile or non-volatile storage medium storing one or more sequences of instructions which when executed by one or more processors, cause the one or more processors to perform the steps of:

automatically generating a consistent user interface for an application program by assisting a user with building an [HTML] hypertext markup language (HTML) user interface page by:

receiving one or more business objects that each define a user action for the application program;

receiving one or more metadata elements for dynamic content generation and defining parameters for the user actions of the one or more business object;

invoking a controller that is communicatively coupled to one or more actions, one or more widgets, and one or more panels;

receiving a first user request from the user through a browser used to interact with the application program and dispatching the user request to one or more of the actions;

wherein the first panel is dynamically generated using the one or more widgets

and comprises a programmed template with a user-interface content; wherein the user-interface content comprises [any] one or more of user controls, programmable

buttons and action buttons;

based on a second user request received from the user through the browser, selecting a first panel from the one or more panels and including the first panel in the HTML user interface page;

the controller determining which of the one or more actions is responsible for acting on the first user request;

Art Unit: 2179

obtaining, using the one or more actions, one or more parameter values from the business objects and dynamically manipulating the one or more parameter values;

using the one or more actions and the business object parameter values, selecting

a first widget from the one or more widgets;

associating the first widget with the first panel selected from the one or more panels, wherein the first widget is arranged into a specified dynamic layout within the first panel; and

generating the specified dynamic layout, at runtime, and presenting to the user, a user-interface utility comprising the HTML user interface page that includes the first widget arranged into the specified dynamic layout within the first panel; wherein the first widget has the capability of representing properties of the business objects as HTML elements.

Claim 14. (currently amended) A computer-readable volatile or non-volatile storage medium as recited in Claim 13, wherein the business object parameters are associated with one of the widgets based on the user request.

Claim 15. (currently amended) A computer-readable volatile or non-volatile storage medium as recited in Claim 13, wherein the application program is a network management application program.

Claim 16. (currently amended) A computer-readable volatile or non-volatile storage medium as recited in Claim 13, wherein receiving one or more business objects that define functions of the application program comprises receiving an [XML] extensible markup language (XML) file that defines the business objects and one or more of the parameters for the business objects.

Claim 17. (currently amended) A computer-readable volatile or non-volatile storage medium as recited in Claim 13, further comprising instructions for performing the step of generating, using the widget, client-side executable program code that performs one or more data validation or access control operations on user input for the user operation.

Claim 18. (currently amended) A computer-readable volatile or non-volatile storage medium as recited in Claim 13, wherein the actions interact with the business objects through service object module interfaces that provide parameter values for the business objects to the actions.

Claim 19. (currently amended) A computer-readable volatile or non-volatile storage medium as recited in Claim 13, further comprising instructions for performing the steps of:

Art Unit: 2179

receiving user input in a field of the user interface that is associated with the widget, wherein the user input is received in HTML elements of an [HTTP] hypertext transfer protocol (HTTP) request from a browser; converting the user input from the HTML elements into one or more programmatic objects having an appropriate data type for use by the application program.

Claim 20. (currently amended) A computer-readable volatile or non-volatile storage medium as recited in Claim 13, further comprising instructions for performing the step of associating a first widget with a second widget, wherein the first widget and second widget are related by a containment hierarchy.

Claim 21. (currently amended d) A computer-readable volatile or non-volatile storage medium as recited in Claim 13, wherein each of the widgets represents one or more properties of the business objects by an HTML element.

Claim 22. (currently amended) A computer-readable volatile or non-volatile storage medium as recited in Claim 13, wherein the step of generating an HTML user interface page that includes the panel further comprises generating an HTML user interface page that includes one or more of [JSP] java server pages (JSP) files, static HTML elements, style sheets, or images.

Claim 23. (Currently Amended) An apparatus comprising: one or more processors;

means for automatically generating a consistent user interface for an application program by assisting a user with building an [HTML] hypertext markup language (HTML) user interface page, the means including:

means for receiving one or more business objects that each define a user action

for the application program;

means for receiving one or more metadata elements for dynamic content generation and defining parameters for the user actions of the one or more business object;

means for invoking a controller that is communicatively coupled to one or more

actions, one or more widgets, and one or more panels;

means for receiving a first user request from the user through a browser used to interact with the application program and dispatching the user request to one or more of the actions, wherein the controller determines which of the one or more actions is responsible for acting on the user request;

means for selecting, based on a second user request received from the user through the browser, a first panel from the one or more panels and including the first panel in the HTML user interface page;

wherein the first panel is dynamically generated using the one or more widgets

Art Unit: 2179

and comprises a programmed template with a user-interface content;
wherein the user-interface content comprises [any] one or more of user controls, programmable

buttons and action buttons;

means for obtaining, using the one or more actions, one or more parameter values
from the business objects and dynamically manipulating the one or more
parameter values;

means for selecting a first widget from the one or more widgets, using the one or
more actions and the business object parameter values;

means for associating the first widget with the first panel selected from the one or
more panels, wherein the first widget is arranged into a specified dynamic
layout within the first panel; and

means for generating the specified dynamic layout, at runtime, and presenting to the user,

a user-interface utility comprising the HTML user interface page that includes the
first widget arranged into the specified dynamic layout with in the first panel;
wherein the first widget has the capability of representing properties of the
business objects as HTML elements.

Claim 26. (currently amended) An apparatus as recited in Claim 23, wherein the means for receiving one
or more business objects that define functions of the application program comprises means for
receiving an [XML] extensible markup language (XML) file that defines the business objects and one or
more of the parameters for the business objects.

Claim 29. (currently amended) An apparatus as recited in Claim 23, further comprising:
means for receiving user input in a field of the user interface that is associated with the
widget, wherein the user input is received in HTML elements of an [HTTP] hypertext transfer protocol
(HTTP) request from a browser; means for converting the user input from the HTML elements into one or
more programmatic objects having an appropriate data type for use by the application
program.

Claim 32. (currently amended) An apparatus as recited in Claim 23, wherein the means for generating an
HTML user interface page that includes the panel further comprises means for generating
an HTML user interface page that includes one or more of [JSP] java server pages (JSP) files, static
HTML elements, style sheets, or images.

Claim 33. (Currently Amended) an apparatus comprising:
a network interface that is coupled to the data network for receiving one or more packet
flows therefrom;
a processor;

Art Unit: 2179

one or more stored sequences of instructions which, when executed by the processor, cause the processor to perform the steps of:
automatically generating a consistent user interface for an application program by assisting a user with building an [HTML] hypertext markup language (HTML) user interface page by: receiving one or more business objects that each define a user action for the

application program;
receiving one or more metadata elements for dynamic content generation and

defining parameters for the user actions of the one or more business

object;
invoking a controller that is communicatively coupled to one or more actions, one

or more widgets, and one or more panels;
receiving a first user request from the user through a browser used to interact with

the application program and dispatching the user request to one or more of

the actions;
based on a second user request received from the user through the browser,

selecting a first panel from the one or more panels and including the first panel in the HTML user interface page;

wherein the first panel is dynamically generated using the one or more widgets

and comprises a programmed template with a user-interface content;
wherein the user-interface content comprises [any] one or more of user controls, programmable

buttons and action buttons.:

the controller determining which of the one or more actions is responsible for

acting on the first user request;

obtaining, using the one or more actions, one or more parameter values from the business objects and dynamically manipulating the one or more parameter values;

using the one or more actions and the business object parameter values, selecting

a first widget from the one or more widgets;

associating the first widget with the first panel selected from the one or more panels, wherein the first widget is arranged into a specified dynamic layout within the first panel; and

generating the specified dynamic layout, at runtime, and presenting to the user, a user-interface utility comprising the HTML user interface page that includes the first widget arranged into the specified dynamic layout within the first panel; wherein the first widget has the capability of representing properties of the business objects as HTML elements.

Claim 36. (currently amended) An apparatus as recited in Claim 33, wherein receiving one or more business objects that define functions of the application program comprises receiving an [XML] extensible markup language (XML) file that defines the business objects and one or more of the parameters for the business objects.

Claim 39. (currently amended) An apparatus as recited in Claim 33, further comprising instructions for performing the steps of: receiving user input in a field of the user interface that is associated with the widget, wherein the user input is received in HTML elements of an [HTTP] hypertext transfer protocol (HTTP) request from a browser; converting the user input from the HTML elements into one or more programmatic objects having an appropriate data type for use by the application program.

Allowable Subject Matter

3. Claims 2-11 and 13-42 are allowed.

The following is an examiner's statement of reasons for allowance: Independent claims 2, 13, 23 and 33, when considered as a whole, are allowable over the prior art of record. Specifically, the prior art of Underwood teaches the limitation: "an image definer that is predefined graphic user interface panel supplied from a vendor to a user that can not be dynamically generated by the user; tools are provided by the image definer to manage images which can be imported to a website". But the claims recite a different combination of limitation: "automatically generating a consistent user interface for an application program by assisting a user with building an HTML user interface page... selecting a first panel from the one or more panels and including the first panel in the HTML user interface page; wherein the first panel is dynamically generated using the one or more widgets and comprises a programmed template with a user-interface

content; and wherein the user interface content comprises one or more of user controls, programmable buttons and action buttons" (limitation A), that is not suggested or shown by Underwood.

The prior art of Underwood2 teaches another combination, "a control panel that is used to install and configure a database", but does not suggest the limitation A.

Further Underwood and Underwood2 teaches the limitation: "an image definer using a control panel to install and configure databases, but the claims recite a different combination of limitation A that is not suggested or shown by Underwood or Underwood2.

The dependent claims further add limitations to the allowable subject matter of the corresponding independent claims; thus are also allowable. Therefore the claims are allowed over the art because the claims differ in scope that is not seen or suggested by the prior art.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Inquires

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Nicholas Augustine whose telephone number is 571-270-1056 and fax is 571-270-2056. The examiner can normally be reached on Monday - Friday: 9:30am- 5:00pm Eastern.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Weilun Lo can be reached on 571-272-4847. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Weilun Lo/
Supervisory Patent Examiner, Art Unit 2179

/Nicholas Augustine/
Examiner
Art Unit 2179
September 7, 2010